

Отчет по практической работе №5.1

Работа с технологиями контейнеризации и ботами Telegram

Цель работы: модернизация уже существующего Telegram бота. Реализация вывода графиков из практической работы №4 и №5.

Ход работы:

Подключаемся к рабочему серверу, активируем виртуальное окружение и устанавливаем необходимые библиотеки (pandas, matplotlib, openpyxl) для создания графиков и диаграмм.

```
(env) student@user-IPMSB-H61:~/220803144$ pip install pandas
Collecting pandas
  Using cached pandas-2.2.3-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (13.1 MB)
Collecting numpy>=1.22.4
  Using cached numpy-2.2.3-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (16.4 MB)
Collecting tzdata>=2022.7
  Using cached tzdata-2025.1-py2.py3-none-any.whl (346 kB)
Collecting pytz>=2020.1
  Using cached pytz-2025.1-py2.py3-none-any.whl (507 kB)
Collecting python-dateutil>=2.8.2
  Using cached python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)
Collecting six>=1.5
  Using cached six-1.17.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: pytz, tzdata, six, numpy, python-dateutil, pandas
Successfully installed numpy-2.2.3 pandas-2.2.3 python-dateutil-2.9.0.post0 pytz-2025.1 six-1.17.0 tzdata-2025.1

[notice] A new release of pip is available: 23.0.1 -> 25.0.1
[notice] To update, run: pip install --upgrade pip
```

Рис 5.1.1

```
(env) student@user-IPMSB-H61:~/220803144$ pip install matplotlib
Collecting matplotlib
  Using cached matplotlib-3.10.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (8.6 MB)
Collecting kiwisolver>=1.3.1
  Using cached kiwisolver-1.4.8-cp310-cp310-manylinux_2_12_x86_64.manylinux2010_x86_64.whl (1.6 MB)
Collecting pillow>=8
  Using cached pillow-11.1.0-cp310-cp310-manylinux_2_28_x86_64.whl (4.5 MB)
Collecting cycler>=0.10
  Using cached cycler-0.12.1-py3-none-any.whl (8.3 kB)
Requirement already satisfied: python-dateutil>=2.7 in ./env/lib/python3.10/site-packages (from matplotlib) (2.9.0.post0)
Collecting packaging>=20.0
  Using cached packaging-24.2-py3-none-any.whl (65 kB)
Collecting contourpy>=1.0.1
  Using cached contourpy-1.3.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (324 kB)
Collecting pyparsing>=2.3.1
  Using cached pyparsing-3.2.1-py3-none-any.whl (107 kB)
Collecting fonttools>=4.22.0
  Using cached fonttools-4.56.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (4.6 MB)
Requirement already satisfied: numpy>=1.23 in ./env/lib/python3.10/site-packages (from matplotlib) (2.2.3)
Requirement already satisfied: six>=1.5 in ./env/lib/python3.10/site-packages (from python-dateutil>=2.7->matplotlib) (1.17.0)
Installing collected packages: pyparsing, pillow, packaging, kiwisolver, fonttools, cycler, contourpy, matplotlib
Successfully installed contourpy-1.3.1 cycler-0.12.1 fonttools-4.56.0 kiwisolver-1.4.8 matplotlib-3.10.1 packaging-24.2 pillow-11.1.0 pyparsing-3.2.1

[notice] A new release of pip is available: 23.0.1 -> 25.0.1
[notice] To update, run: pip install --upgrade pip
```

Рис 5.1.2

```
(env) student@user-IPMSB-H61:~/220803144$ pip install openpyxl
Collecting openpyxl
  Using cached openpyxl-3.1.5-py2.py3-none-any.whl (250 kB)
Collecting et-xmlfile
  Using cached et_xmlfile-2.0.0-py3-none-any.whl (18 kB)
Installing collected packages: et-xmlfile, openpyxl
Successfully installed et-xmlfile-2.0.0 openpyxl-3.1.5

[notice] A new release of pip is available: 23.0.1 -> 25.0.1
[notice] To update, run: pip install --upgrade pip
```

Рис 5.1.3

Далее необходимо изменить код, чтоб бот по командам /command1 и /command2 выводил не Ок и Oks, а графики из практической работы №4 и №5. Код для графиков я брал из соответствующих практических работ.

```
import telepot
import time
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

def handle(msg):
    chat_id = msg['chat']['id']
    command = msg['text']
    print('Got command: %s' % command)
    print('From : %s' % chat_id)

    if command == '/command1':
        x = np.linspace(-10, 10, 100)
        a = np.random.random()
        y = a * np.sin(x)
        plt.plot(x, y, marker="x")
        plt.title('График синусоиды')
        plt.xlabel('x')
        plt.ylabel('y')
        plt.grid()
        plt.savefig('graph1.png')
        plt.close()

        bot.sendPhoto(chat_id, photo=open('graph1.png', 'rb'))

    elif command == '/command2':
        df_can = pd.read_excel(
            'https://s3-api.us-gio.objectstorage.softlayer.net/cf-courses-data/CognitiveClass/DV0101EN/labs/Data_Files/Canada.xlsx',
            sheet_name='Canada by Citizenship',
            skiprows=range(20),
            skipfooter=2)

        df_can.drop(['AREA', 'REG', 'DEV', 'Type', 'Coverage'], axis=1, inplace=True)
        df_can.rename(columns={'OdName': 'Country', 'AreaName': 'Continent', 'RegName': 'Region'}, inplace=True)
        all(isinstance(column, str) for column in df_can.columns)
        df_can.columns = list(map(str, df_can.columns))
        all(isinstance(column, str) for column in df_can.columns)
        df_can.set_index('Country', inplace=True)
        df_can['Total'] = df_can.select_dtypes(include='number').sum(axis=1)
        years = list(map(str, range(1980, 2014)))
        df_can.sort_values(['Total'], ascending=False, axis=0, inplace=True)
        df_top5 = df_can.head()
        df_top5 = df_top5[years].transpose()
```

Рис 5.1.4

```
plt.figure(figsize=(8, 5))
df_can['2013'].plot(kind='hist', bins=30)
plt.title('Гистограмма иммиграции из 195 стран в 2013 году')
plt.ylabel('Количество стран')
plt.xlabel('Количество иммигрантов')
plt.savefig('histogram.png')
plt.close()

bot.sendPhoto(chat_id, photo=open('histogram.png', 'rb'))

bot = telepot.Bot('7909336616:AAHCSTdcV9WD1NrYWL8xCQUYDajJ7YjkV8')
bot.message_loop(handle)

print('I am listening ...')

while True:
    time.sleep(10)
```

Рис 5.1.5

Проверяем работу бота.

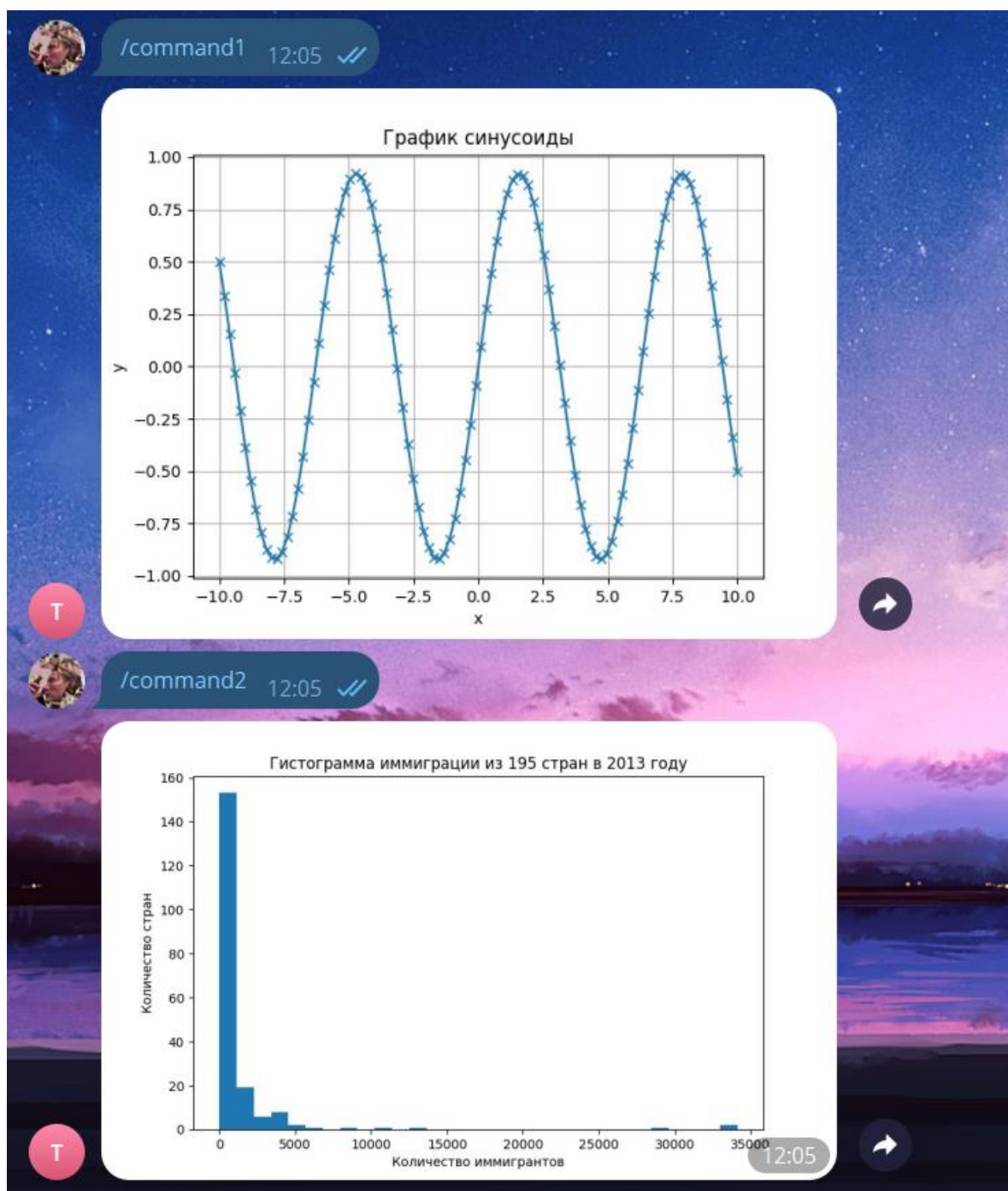


Рис 5.1.6

Теперь необходимо создать `docker_image` и запустить `docker` контейнер. Обновляем файл `requirements` и создаем `docker_image`.

```
telepot==12.7
matplotlib
numpy
pandas
```

Рис 5.1.7

```
student@user-IPMSB-H61:~/220803144$ docker build -t 220803144-graph-histogram .
[+] Building 147.2s (13/13) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile
=> [internal] load metadata for docker.io/library/python:3.10-slim
=> [internal] load metadata for docker.io/library/python:3.10
=> [internal] load .dockerignore
=> transferring context: 20B
=> CACHED [builder 1/3] FROM docker.io/library/python:3.10-slim@sha256:c7bc-d7054564482c80ee8cd0d0e114458aac59ea01f6691fa715207ea0e
=> [stage-1 1/4] FROM docker.io/library/python:3.10-slim@sha256:c7bc-d7054564482c80ee8cd0d0e114458aac59ea01f6691fa715207ea0e
=> [internal] load build context
=> transferring context: 2.38kB
=> [builder 2/3] COPY requirements.txt
=> [builder 3/3] RUN pip install --user -r requirements.txt
=> CACHED [stage-1 2/4] WORKDIR /code
=> [stage-1 3/4] COPY --from=builder /root/.local /root/.local
=> [stage-1 4/4] COPY ./bot.py
=> exporting to image
=> exporting layers
=> writing image sha256:8a7bf77ba779a44dc1a673c896dbcc9ed7c9228eb208c876f2304b4378a6a7c
=> naming to docker.io/library/220803144-graph-histogram
```

Рис 5.1.8

Просматриваем логи на наличие ошибок.

```
student@user-IPMSB-H61:~/220803144$ docker logs 7b58433a197a49affa0c57aa159337a8f5a1cc121d0e6dd9754d235c981bd5b8
I am listening ...
Got command: /command1
From : 1017881787
Got command: /command2
From : 1017881787
```

Рис 5.1.9

Ошибок нет. Сохраняем `docker_image` на сервере и на ПК.

Вывод: в результате практической работы был усовершенствован созданный Telegram bot. В окружение были загружены необходимые библиотеки. Код бота был изменен так, чтоб на команды `/command1` и `/command2` бот выводил фото графика или диаграммы. Проблемой при реализации практической стало то, что контейнер со старым ботом уже был запущен и новый бот конфликтовал при запуске со старым. Необходимо было убрать контейнер со старым ботом из работы в фоне.